# Project 1: Solar Flares


## By Alejandro Rigau


### 3/4/2021

---

# Part 1: Data scraping and preparation

## Step 1: Scrape your competitor's data

Here I use the html parser from BeautifulSoup to get the html from the page. From there, I find the table, and then load it to a pandas dataframe.

In [1]:
```
import requests
import pandas as pd
import numpy as np
import datetime #for step 2
from bs4 import BeautifulSoup
```

In [2]:
```
url = 'https://cmsc320.github.io/files/top-50-solar-flares.html'
page = requests.get(url)
soup = BeautifulSoup(page.text, 'html.parser')
```

In [3]:
```
# print(soup.prettify())
```

In [4]:
```
table = soup.find('table')
# print(table)
```

In [5]:
```
space = pd.read_html(str(table))[0]
space.columns = ["rank", "x_classification", "date", "region", "start_time", "maximum_time", "end_time", "movie"]
space
```

Out[5]:

| | rank | x_classification | date | region | start_time | maximum_time | end_time | movie |
|---|---|---|---|---|---|---|---|---|
| **0** | 1 | X28+ | 2003/11/04 | 486 | 19:29 | 19:53 | 20:06 | MovieView archive |
| **1** | 2 | X20+ | 2001/04/02 | 9393 | 21:32 | 21:51 | 22:03 | MovieView archive |
| **2** | 3 | X17.2+ | 2003/10/28 | 486 | 09:51 | 11:10 | 11:24 | MovieView archive |
| **3** | 4 | X17+ | 2005/09/07 | 808 | 17:17 | 17:40 | 18:03 | MovieView archive |
| **4** | 5 | X14.4 | 2001/04/15 | 9415 | 13:19 | 13:50 | 13:55 | MovieView archive |
| **5** | 6 | X10 | 2003/10/29 | 486 | 20:37 | 20:49 | 21:01 | MovieView archive |
| **6** | 7 | X9.4 | 1997/11/06 | 8100 | 11:49 | 11:55 | 12:01 | MovieView archive |
| **7** | 8 | X9.3 | 2017/09/06 | 2673 | 11:53 | 12:02 | 12:10 | MovieView archive |
| **8** | 9 | X9 | 2006/12/05 | 930 | 10:18 | 10:35 | 10:45 | MovieView archive |
| **9** | 10 | X8.3 | 2003/11/02 | 486 | 17:03 | 17:25 | 17:39 | MovieView archive |
| **10** | 11 | X8.2 | 2017/09/10 | 2673 | 15:35 | 16:06 | 16:31 | MovieView archive |
| **11** | 12 | X7.1 | 2005/01/20 | 720 | 06:36 | 07:01 | 07:26 | MovieView archive |
| **12** | 13 | X6.9 | 2011/08/09 | 1263 | 07:48 | 08:05 | 08:08 | MovieView archive |
| **13** | 14 | X6.5 | 2006/12/06 | 930 | 18:29 | 18:47 | 19:00 | MovieView archive |
| **14** | 15 | X6.2 | 2005/09/09 | 808 | 19:13 | 20:04 | 20:36 | MovieView archive |
| **15** | 16 | X6.2 | 2001/12/13 | 9733 | 14:20 | 14:30 | 14:35 | MovieView archive |
| **16** | 17 | X5.7 | 2000/07/14 | 9077 | 10:03 | 10:24 | 10:43 | MovieView archive |
| **17** | 18 | X5.6 | 2001/04/06 | 9415 | 19:10 | 19:21 | 19:31 | MovieView archive |
| **18** | 19 | X5.4 | 2012/03/07 | 1429 | 00:02 | 00:24 | 00:40 | MovieView archive |
| **19** | 20 | X5.4 | 2005/09/08 | 808 | 20:52 | 21:06 | 21:17 | MovieView archive |
| **20** | 21 | X5.4 | 2003/10/23 | 486 | 08:19 | 08:35 | 08:49 | MovieView archive |
| **21** | 22 | X5.3 | 2001/08/25 | 9591 | 16:23 | 16:45 | 17:04 | MovieView archive |
| **22** | 23 | X4.9 | 2014/02/25 | 1990 | 00:39 | 00:49 | 01:03 | MovieView archive |
| **23** | 24 | X4.9 | 1998/08/18 | 8307 | 22:10 | 22:19 | 22:28 | View archive |
| **24** | 25 | X4.8 | 2002/07/23 | 39 | 00:18 | 00:35 | 00:47 | MovieView archive |
| **25** | 26 | X4 | 2000/11/26 | 9236 | 16:34 | 16:48 | 16:56 | MovieView archive |
| **26** | 27 | X3.9 | 2003/11/03 | 488 | 09:43 | 09:55 | 10:19 | MovieView archive |
| **27** | 28 | X3.9 | 1998/08/19 | 8307 | 21:35 | 21:45 | 21:50 | View archive |
| **28** | 29 | X3.8 | 2005/01/17 | 720 | 06:59 | 09:52 | 10:07 | MovieView archive |
| **29** | 30 | X3.7 | 1998/11/22 | 8384 | 06:30 | 06:42 | 06:49 | MovieView archive |
| **30** | 31 | X3.6 | 2005/09/09 | 808 | 09:42 | 09:59 | 10:08 | MovieView archive |
| **31** | 32 | X3.6 | 2004/07/16 | 649 | 13:49 | 13:55 | 14:01 | MovieView archive |
| **32** | 33 | X3.6 | 2003/05/28 | 365 | 00:17 | 00:27 | 00:39 | MovieView archive |
| **33** | 34 | X3.4 | 2006/12/13 | 930 | 02:14 | 02:40 | 02:57 | MovieView archive |
| **34** | 35 | X3.4 | 2001/12/28 | 9767 | 20:02 | 20:45 | 21:32 | MovieView archive |
| **35** | 36 | X3.3 | 2013/11/05 | 1890 | 22:07 | 22:12 | 22:15 | MovieView archive |
| **36** | 37 | X3.3 | 2002/07/20 | 39 | 21:04 | 21:30 | 21:54 | MovieView archive |
| **37** | 38 | X3.3 | 1998/11/28 | 8395 | 04:54 | 05:52 | 06:13 | MovieView archive |
| **38** | 39 | X3.2 | 2013/05/14 | 1748 | 00:00 | 01:11 | 01:20 | MovieView archive |
| **39** | 40 | X3.1 | 2014/10/24 | 2192 | 21:07 | 21:41 | 22:13 | MovieView archive |
| **40** | 41 | X3.1 | 2002/08/24 | 69 | 00:49 | 01:12 | 01:31 | MovieView archive |
| **41** | 42 | X3 | 2002/07/15 | 30 | 19:59 | 20:08 | 20:14 | MovieView archive |
| **42** | 43 | X2.8 | 2013/05/13 | 1748 | 15:48 | 16:05 | 16:16 | MovieView archive |
| **43** | 44 | X2.8 | 2001/12/11 | 9733 | 07:58 | 08:08 | 08:14 | MovieView archive |
| **44** | 45 | X2.8 | 1998/08/18 | 8307 | 08:14 | 08:24 | 08:32 | View archive |
| **45** | 46 | X2.7 | 2015/05/05 | 2339 | 22:05 | 22:11 | 22:15 | MovieView archive |
| **46** | 47 | X2.7 | 2003/11/03 | 488 | 01:09 | 01:30 | 01:45 | MovieView archive |
| **47** | 48 | X2.7 | 1998/05/06 | 8210 | 07:58 | 08:09 | 08:20 | MovieView archive |
| **48** | 49 | X2.6 | 2005/01/15 | 720 | 22:25 | 23:02 | 23:31 | MovieView archive |
| **49** | 50 | X2.6 | 2001/09/24 | 9632 | 09:32 | 10:38 | 11:09 | MovieView archive |

# Step 2: Tidy the top 50 solar flare data

## 2.1 Dropping movie

In [6]:
```
space = space.drop('movie', 1)
space.head(2)
```

Out[6]:

| | rank | x_classification | date | region | start_time | maximum_time | end_time |
|---|---|---|---|---|---|---|---|
| **0** | 1 | X28+ | 2003/11/04 | 486 | 19:29 | 19:53 | 20:06 |
| **1** | 2 | X20+ | 2001/04/02 | 9393 | 21:32 | 21:51 | 22:03 |

## 2.2 and 2.3 Date time combination

Here I add the strings from date and the clock time and use the to_datetime function to automatically generate the correct format and then drop the unnecessary columns.

In [7]:
```
space['start_datetime'] = pd.to_datetime(space['date']+ " " +space['start_time'])
space['max_datetime'] = pd.to_datetime(space['date']+ " " +space['maximum_time'])
space['end_datetime'] = pd.to_datetime(space['date']+ " " +space['end_time'])
space = space.drop('start_time', 1)
space = space.drop('maximum_time', 1)
space = space.drop('end_time', 1)
space = space.drop('date', 1)

temp = space['region'] # basically moving it to the end
space = space.drop('region', 1)
space['region'] = temp
```

## 2.4 drop nan

Here I use standard pandas function to replace all '-' in the dataset. Lastly, I print out my final dataframe for this part.

In [8]:
```
space = space.replace('-', np.nan) #step 2.4
```

In [9]:
```
space
```

Out[9]:

| | rank | x_classification | start_datetime | max_datetime | end_datetime | region |
|---|---|---|---|---|---|---|
| **0** | 1 | X28+ | 2003-11-04 19:29:00 | 2003-11-04 19:53:00 | 2003-11-04 20:06:00 | 486 |
| **1** | 2 | X20+ | 2001-04-02 21:32:00 | 2001-04-02 21:51:00 | 2001-04-02 22:03:00 | 9393 |
| **2** | 3 | X17.2+ | 2003-10-28 09:51:00 | 2003-10-28 11:10:00 | 2003-10-28 11:24:00 | 486 |
| **3** | 4 | X17+ | 2005-09-07 17:17:00 | 2005-09-07 17:40:00 | 2005-09-07 18:03:00 | 808 |
| **4** | 5 | X14.4 | 2001-04-15 13:19:00 | 2001-04-15 13:50:00 | 2001-04-15 13:55:00 | 9415 |
| **5** | 6 | X10 | 2003-10-29 20:37:00 | 2003-10-29 20:49:00 | 2003-10-29 21:01:00 | 486 |
| **6** | 7 | X9.4 | 1997-11-06 11:49:00 | 1997-11-06 11:55:00 | 1997-11-06 12:01:00 | 8100 |
| **7** | 8 | X9.3 | 2017-09-06 11:53:00 | 2017-09-06 12:02:00 | 2017-09-06 12:10:00 | 2673 |
| **8** | 9 | X9 | 2006-12-05 10:18:00 | 2006-12-05 10:35:00 | 2006-12-05 10:45:00 | 930 |
| **9** | 10 | X8.3 | 2003-11-02 17:03:00 | 2003-11-02 17:25:00 | 2003-11-02 17:39:00 | 486 |
| **10** | 11 | X8.2 | 2017-09-10 15:35:00 | 2017-09-10 16:06:00 | 2017-09-10 16:31:00 | 2673 |
| **11** | 12 | X7.1 | 2005-01-20 06:36:00 | 2005-01-20 07:01:00 | 2005-01-20 07:26:00 | 720 |
| **12** | 13 | X6.9 | 2011-08-09 07:48:00 | 2011-08-09 08:05:00 | 2011-08-09 08:08:00 | 1263 |
| **13** | 14 | X6.5 | 2006-12-06 18:29:00 | 2006-12-06 18:47:00 | 2006-12-06 19:00:00 | 930 |
| **14** | 15 | X6.2 | 2005-09-09 19:13:00 | 2005-09-09 20:04:00 | 2005-09-09 20:36:00 | 808 |
| **15** | 16 | X6.2 | 2001-12-13 14:20:00 | 2001-12-13 14:30:00 | 2001-12-13 14:35:00 | 9733 |
| **16** | 17 | X5.7 | 2000-07-14 10:03:00 | 2000-07-14 10:24:00 | 2000-07-14 10:43:00 | 9077 |
| **17** | 18 | X5.6 | 2001-04-06 19:10:00 | 2001-04-06 19:21:00 | 2001-04-06 19:31:00 | 9415 |
| **18** | 19 | X5.4 | 2012-03-07 00:02:00 | 2012-03-07 00:24:00 | 2012-03-07 00:40:00 | 1429 |
| **19** | 20 | X5.4 | 2005-09-08 20:52:00 | 2005-09-08 21:06:00 | 2005-09-08 21:17:00 | 808 |
| **20** | 21 | X5.4 | 2003-10-23 08:19:00 | 2003-10-23 08:35:00 | 2003-10-23 08:49:00 | 486 |
| **21** | 22 | X5.3 | 2001-08-25 16:23:00 | 2001-08-25 16:45:00 | 2001-08-25 17:04:00 | 9591 |
| **22** | 23 | X4.9 | 2014-02-25 00:39:00 | 2014-02-25 00:49:00 | 2014-02-25 01:03:00 | 1990 |
| **23** | 24 | X4.9 | 1998-08-18 22:10:00 | 1998-08-18 22:19:00 | 1998-08-18 22:28:00 | 8307 |
| **24** | 25 | X4.8 | 2002-07-23 00:18:00 | 2002-07-23 00:35:00 | 2002-07-23 00:47:00 | 39 |
| **25** | 26 | X4 | 2000-11-26 16:34:00 | 2000-11-26 16:48:00 | 2000-11-26 16:56:00 | 9236 |
| **26** | 27 | X3.9 | 2003-11-03 09:43:00 | 2003-11-03 09:55:00 | 2003-11-03 10:19:00 | 488 |
| **27** | 28 | X3.9 | 1998-08-19 21:35:00 | 1998-08-19 21:45:00 | 1998-08-19 21:50:00 | 8307 |
| **28** | 29 | X3.8 | 2005-01-17 06:59:00 | 2005-01-17 09:52:00 | 2005-01-17 10:07:00 | 720 |
| **29** | 30 | X3.7 | 1998-11-22 06:30:00 | 1998-11-22 06:42:00 | 1998-11-22 06:49:00 | 8384 |
| **30** | 31 | X3.6 | 2005-09-09 09:42:00 | 2005-09-09 09:59:00 | 2005-09-09 10:08:00 | 808 |
| **31** | 32 | X3.6 | 2004-07-16 13:49:00 | 2004-07-16 13:55:00 | 2004-07-16 14:01:00 | 649 |
| **32** | 33 | X3.6 | 2003-05-28 00:17:00 | 2003-05-28 00:27:00 | 2003-05-28 00:39:00 | 365 |
| **33** | 34 | X3.4 | 2006-12-13 02:14:00 | 2006-12-13 02:40:00 | 2006-12-13 02:57:00 | 930 |
| **34** | 35 | X3.4 | 2001-12-28 20:02:00 | 2001-12-28 20:45:00 | 2001-12-28 21:32:00 | 9767 |
| **35** | 36 | X3.3 | 2013-11-05 22:07:00 | 2013-11-05 22:12:00 | 2013-11-05 22:15:00 | 1890 |
| **36** | 37 | X3.3 | 2002-07-20 21:04:00 | 2002-07-20 21:30:00 | 2002-07-20 21:54:00 | 39 |
| **37** | 38 | X3.3 | 1998-11-28 04:54:00 | 1998-11-28 05:52:00 | 1998-11-28 06:13:00 | 8395 |
| **38** | 39 | X3.2 | 2013-05-14 00:00:00 | 2013-05-14 01:11:00 | 2013-05-14 01:20:00 | 1748 |
| **39** | 40 | X3.1 | 2014-10-24 21:07:00 | 2014-10-24 21:41:00 | 2014-10-24 22:13:00 | 2192 |
| **40** | 41 | X3.1 | 2002-08-24 00:49:00 | 2002-08-24 01:12:00 | 2002-08-24 01:31:00 | 69 |
| **41** | 42 | X3 | 2002-07-15 19:59:00 | 2002-07-15 20:08:00 | 2002-07-15 20:14:00 | 30 |
| **42** | 43 | X2.8 | 2013-05-13 15:48:00 | 2013-05-13 16:05:00 | 2013-05-13 16:16:00 | 1748 |
| **43** | 44 | X2.8 | 2001-12-11 07:58:00 | 2001-12-11 08:08:00 | 2001-12-11 08:14:00 | 9733 |
| **44** | 45 | X2.8 | 1998-08-18 08:14:00 | 1998-08-18 08:24:00 | 1998-08-18 08:32:00 | 8307 |
| **45** | 46 | X2.7 | 2015-05-05 22:05:00 | 2015-05-05 22:11:00 | 2015-05-05 22:15:00 | 2339 |
| **46** | 47 | X2.7 | 2003-11-03 01:09:00 | 2003-11-03 01:30:00 | 2003-11-03 01:45:00 | 488 |
| **47** | 48 | X2.7 | 1998-05-06 07:58:00 | 1998-05-06 08:09:00 | 1998-05-06 08:20:00 | 8210 |
| **48** | 49 | X2.6 | 2005-01-15 22:25:00 | 2005-01-15 23:02:00 | 2005-01-15 23:31:00 | 720 |
| **49** | 50 | X2.6 | 2001-09-24 09:32:00 | 2001-09-24 10:38:00 | 2001-09-24 11:09:00 | 9632 |

## Step 3: Scrape the NASA data

Here we cant use BeautifulSoup to find the table tag but I can use it to get the text and separate it later by each line and by spaces. I also remove the header part of the table and the footer. Lastly, I go thorough each line and I split each string into their columns.

In [10]:
```
url = 'http://www.hcbravo.org/IntroDataSci/misc/waves_type2.html'
page = requests.get(url)
soup = BeautifulSoup(page.text, 'html.parser')
```

In [11]:
```
table = soup.find('pre')
table = table.text.split('\n')
table = table[12:-3] # remove unwanted parts of the table (top and bottom)
for row_index in range(len(table)):
    table[row_index] = table[row_index].split()[:14]
```

In [12]:
```
nasa = pd.DataFrame(table)
nasa.columns = ['start_date', 'start_time', 'end_date', 'end_time', 'start_frequency', 'end_frequency', 'flare_location', 'flare_regi
on','flare_classification', 'cme_date', 'cme_time', 'cme_angle', 'cme_width', 'cme_speed']
nasa
```

Out[12]:

| | start_date | start_time | end_date | end_time | start_frequency | end_frequency | flare_location | flare_region | flare_classification | cme_date | cme_time | cme_angle | cm |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1997/04/01 | 14:00 | 04/01 | 14:15 | 8000 | 4000 | S25E16 | 8026 | M1.3 | 04/01 | 15:18 | 74 | |
| **1** | 1997/04/07 | 14:30 | 04/07 | 17:30 | 11000 | 1000 | S28E19 | 8027 | C6.8 | 04/07 | 14:27 | Halo | |
| **2** | 1997/05/12 | 05:15 | 05/14 | 16:00 | 12000 | 80 | N21W08 | 8038 | C1.3 | 05/12 | 05:30 | Halo | |
| **3** | 1997/05/21 | 20:20 | 05/21 | 22:00 | 5000 | 500 | N05W12 | 8040 | M1.3 | 05/21 | 21:00 | 263 | |
| **4** | 1997/09/23 | 21:53 | 09/23 | 22:16 | 6000 | 2000 | S29E25 | 8088 | C1.4 | 09/23 | 22:02 | 133 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **477** | 2014/12/13 | 14:27 | 12/13 | 14:51 | 14000 | 3900 | W90b | ----- | ---- | 12/13 | 14:24 | Halo | |
| **478** | 2014/12/17 | 04:09 | 12/17 | 04:19 | 2900 | 2100 | S11E33 | 12241 | M1.1 | 12/17 | 02:00 | 107 | |
| **479** | 2014/12/17 | 05:00 | 12/17 | 05:09 | 14000 | 11500 | S20E09 | 12242 | M8.7 | 12/17 | 05:00 | Halo | |
| **480** | 2014/12/18 | 22:31 | 12/18 | 22:54 | 5100 | 1300 | S11E15 | 12241 | M6.9 | 12/19 | 01:04 | Halo | |
| **481** | 2014/12/21 | 12:05 | 12/21 | 12:28 | 14000 | 7400 | S14W25 | 12241 | M1.0 | 12/21 | 12:12 | Halo | |

482 rows × 14 columns

## Step 4: Tidy the NASA the table

### 4.1 Replace with NaN

I use regex in this part to find all unfilled datapoints and I replace it with a NaN string.

In [13]:
```
# Remove bad stuff and replace with NaN
nasa = nasa.replace(['-+'], ['NaN'], regex=True)
nasa = nasa.replace(['\?+'], ['NaN'], regex=True)
```

### 4.2 Create Halo column

Using a map function, I go through each row of the cme_angle column and I add to the new column the boolean value.

In [14]:
```
nasa['is_halo'] = nasa['cme_angle'].map(lambda x: True if x == "Halo" else False)
# verify that is_halo worked
nasa.loc[nasa['is_halo'] == True].head()
```

Out[14]:

| | start_date | start_time | end_date | end_time | start_frequency | end_frequency | flare_location | flare_region | flare_classification | cme_date | cme_time | cme_angle | cm |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 1997/04/07 | 14:30 | 04/07 | 17:30 | 11000 | 1000 | S28E19 | 8027 | C6.8 | 04/07 | 14:27 | Halo | |
| **2** | 1997/05/12 | 05:15 | 05/14 | 16:00 | 12000 | 80 | N21W08 | 8038 | C1.3 | 05/12 | 05:30 | Halo | |
| **7** | 1997/11/04 | 06:00 | 11/05 | 04:30 | 14000 | 100 | S14W33 | 8100 | X2.1 | 11/04 | 06:10 | Halo | |
| **8** | 1997/11/06 | 12:20 | 11/07 | 08:30 | 14000 | 100 | S18W63 | 8100 | X9.4 | 11/06 | 12:10 | Halo | |
| **11** | 1998/01/25 | 15:03 | 01/25 | 15:18 | 14000 | 10000 | N21E25 | 8141 | C1.1 | 01/25 | 15:26 | Halo | |

### 4.3 Create width_lower_bound column

I do the same thing with the cme_width verifying if the '>' symbol is present.

In [15]:
```
nasa['width_lower_bound'] = nasa['cme_width'].map(lambda x: True if '>' in x else False)
# verifying that width_lower_bound worked
nasa.loc[nasa['width_lower_bound'] == True].head()
```

Out[15]:

| | start_date | start_time | end_date | end_time | start_frequency | end_frequency | flare_location | flare_region | flare_classification | cme_date | cme_time | cme_angle | cm |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **21** | 1998/05/11 | 21:40 | 05/11 | 22:00 | 10000 | 1000 | N32W90 | 8214 | B6.6 | 05/11 | 21:55 | 208 | |
| **30** | 1998/11/06 | 03:00 | 11/06 | 05:30 | 5000 | 1000 | BACK | NaN | NaN | 11/06 | 02:18 | 159 | |
| **39** | 1999/06/11 | 11:45 | 06/11 | 17:00 | 14000 | 400 | N38E90 | NaN | C8.8 | 06/11 | 11:26 | 35 | |
| **41** | 1999/06/23 | 05:50 | 06/23 | 07:10 | 12000 | 2000 | BACK | NaN | NaN | 06/23 | 06:06 | 264 | |
| **43** | 1999/06/28 | 21:03 | 06/28 | 21:10 | 3500 | 1500 | N22W44 | 8592 | C3.5 | 06/28 | 21:30 | 336 | |

## 4.4 Combine date and time columns

Like previously, I add the date and time columns together and pass them through the datetime function that automatically places it in the correct format. Afterwards, I add the new columns into the dataframe and I reorder them to match the requirements.

In [16]:
```
# fix time
nasa = nasa.replace('24:00', '00:00')

temp3 = pd.to_datetime(nasa['start_date'].map(lambda x: x[0:4])+ "/" + nasa['cme_date']+ " " +nasa['cme_time'], errors='coerce')
temp = pd.to_datetime(nasa['start_date'].map(lambda x: x[0:4])+ "/" + nasa['end_date']+ " " +nasa['end_time'])
temp2 = pd.to_datetime(nasa['start_date']+ " " +nasa['start_time'])

nasa.insert(loc=0, column='cme_datetime', value=temp3)
nasa.insert(loc=0, column='end_datetime', value=temp)
nasa.insert(loc=0, column='start_datetime', value=temp2)


nasa = nasa.drop(['start_date','start_time','end_date','end_time','cme_date','cme_time',], 1)

#set column order
nasa = nasa[['start_datetime', 'end_datetime', 'start_frequency',
       'end_frequency', 'flare_location', 'flare_region', 'cme_datetime',
       'flare_classification', 'cme_angle',
       'cme_width', 'cme_speed', 'is_halo', 'width_lower_bound']]
nasa.columns = ['start_datetime', 'end_datetime', 'start_frequency',
       'end_frequency', 'flare_location', 'flare_region', 'cme_datetime',
       'flare_classification', 'cme_angle',
       'cme_width', 'cme_speed', 'is_halo', 'width_lower_bound']
```

### Final Data

In [17]:
```
nasa
```

Out[17]:

| | start_datetime | end_datetime | start_frequency | end_frequency | flare_location | flare_region | cme_datetime | flare_classification | cme_angle | cme_width | cme_speed |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1997-04-01 14:00:00 | 1997-04-01 14:15:00 | 8000 | 4000 | S25E16 | 8026 | 1997-04-01 15:18:00 | M1.3 | 74 | 79 | 312 |
| **1** | 1997-04-07 14:30:00 | 1997-04-07 17:30:00 | 11000 | 1000 | S28E19 | 8027 | 1997-04-07 14:27:00 | C6.8 | Halo | 360 | 878 |
| **2** | 1997-05-12 05:15:00 | 1997-05-14 16:00:00 | 12000 | 80 | N21W08 | 8038 | 1997-05-12 05:30:00 | C1.3 | Halo | 360 | 464 |
| **3** | 1997-05-21 20:20:00 | 1997-05-21 22:00:00 | 5000 | 500 | N05W12 | 8040 | 1997-05-21 21:00:00 | M1.3 | 263 | 165 | 296 |
| **4** | 1997-09-23 21:53:00 | 1997-09-23 22:16:00 | 6000 | 2000 | S29E25 | 8088 | 1997-09-23 22:02:00 | C1.4 | 133 | 155 | 712 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **477** | 2014-12-13 14:27:00 | 2014-12-13 14:51:00 | 14000 | 3900 | W90b | NaN | 2014-12-13 14:24:00 | NaN | Halo | 360 | 2222 |
| **478** | 2014-12-17 04:09:00 | 2014-12-17 04:19:00 | 2900 | 2100 | S11E33 | 12241 | 2014-12-17 02:00:00 | M1.1 | 107 | 108 | 869 |
| **479** | 2014-12-17 05:00:00 | 2014-12-17 05:09:00 | 14000 | 11500 | S20E09 | 12242 | 2014-12-17 05:00:00 | M8.7 | Halo | 360 | 587 |
| **480** | 2014-12-18 22:31:00 | 2014-12-18 22:54:00 | 5100 | 1300 | S11E15 | 12241 | 2014-12-19 01:04:00 | M6.9 | Halo | 360 | 1195 |
| **481** | 2014-12-21 12:05:00 | 2014-12-21 12:28:00 | 14000 | 7400 | S14W25 | 12241 | 2014-12-21 12:12:00 | M1.0 | Halo | 360 | 669 |

482 rows × 13 columns

# Part 2: Analysis

Now that I have the data from both sites, I will use it to answer the questions.

In [18]:
```
space.head()
```

Out[18]:

| | rank | x_classification | start_datetime | max_datetime | end_datetime | region |
|---|---|---|---|---|---|---|
| **0** | 1 | X28+ | 2003-11-04 19:29:00 | 2003-11-04 19:53:00 | 2003-11-04 20:06:00 | 486 |
| **1** | 2 | X20+ | 2001-04-02 21:32:00 | 2001-04-02 21:51:00 | 2001-04-02 22:03:00 | 9393 |
| **2** | 3 | X17.2+ | 2003-10-28 09:51:00 | 2003-10-28 11:10:00 | 2003-10-28 11:24:00 | 486 |
| **3** | 4 | X17+ | 2005-09-07 17:17:00 | 2005-09-07 17:40:00 | 2005-09-07 18:03:00 | 808 |
| **4** | 5 | X14.4 | 2001-04-15 13:19:00 | 2001-04-15 13:50:00 | 2001-04-15 13:55:00 | 9415 |

In [19]:
```
nasa.head()
```

Out[19]:

| | start_datetime | end_datetime | start_frequency | end_frequency | flare_location | flare_region | cme_datetime | flare_classification | cme_angle | cme_width | cme_speed | i |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1997-04-01 14:00:00 | 1997-04-01 14:15:00 | 8000 | 4000 | S25E16 | 8026 | 1997-04-01 15:18:00 | M1.3 | 74 | 79 | 312 | |
| **1** | 1997-04-07 14:30:00 | 1997-04-07 17:30:00 | 11000 | 1000 | S28E19 | 8027 | 1997-04-07 14:27:00 | C6.8 | Halo | 360 | 878 | |
| **2** | 1997-05-12 05:15:00 | 1997-05-14 16:00:00 | 12000 | 80 | N21W08 | 8038 | 1997-05-12 05:30:00 | C1.3 | Halo | 360 | 464 | |
| **3** | 1997-05-21 20:20:00 | 1997-05-21 22:00:00 | 5000 | 500 | N05W12 | 8040 | 1997-05-21 21:00:00 | M1.3 | 263 | 165 | 296 | |
| **4** | 1997-09-23 21:53:00 | 1997-09-23 22:16:00 | 6000 | 2000 | S29E25 | 8088 | 1997-09-23 22:02:00 | C1.4 | 133 | 155 | 712 | |

## Question 1: Replication

Here I do my best to replicate the solar flare table found in SpaceWeatherLive. I first filter by flare classifications that contain the letter X, which are the top ones. Then I remove the X and I turn the column into a float so I can finally sort the values in descending order. Later, I drop unnecessary columns, moved a few columns around and renamed them, and finally added the X to the string. This gives us a really similar representation of the data in SpaceWeatherLive. The only difference in the data is that there are some extra solar flares in the NASA dataset but there are also a few missing items in the NASA dataset.

In [20]:
```
replica = nasa[nasa['flare_classification'].str.contains("X")]
replica['flare_classification'] = replica['flare_classification'].map(lambda y: y.replace("X", ""))
replica['flare_classification'] = replica['flare_classification'].astype(float)
replica = replica.sort_values('flare_classification', ascending=False)

nasa_top_50 = replica.head(50) # for later use in Question 3
replica.head(1)
```

Out[20]:

| | start_datetime | end_datetime | start_frequency | end_frequency | flare_location | flare_region | cme_datetime | flare_classification | cme_angle | cme_width | cme_speed |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **242** | 2003-11-04 20:00:00 | 2003-11-04 | 10000 | 200 | S19W83 | 10486 | 2003-11-04 19:54:00 | 28.0 | Halo | 360 | 2657 |

In [21]:

```
replica = replica.drop(['start_frequency', 'end_frequency', 'flare_location', 'flare_location', 'cme_angle', 'cme_width', 'cme_speed'
, 'is_halo', 'width_lower_bound'], 1)
replica = replica.reset_index(drop=True)
replica = replica.reset_index()
replica['index'] = replica.index + 1
replica = replica[['index', 'flare_classification', 'start_datetime', 'cme_datetime', 'end_datetime', 'flare_region']]
replica.columns = ['rank', 'x_classification', 'start_datetime', 'max_datetime', 'end_datetime','region']
replica = replica.head(50)
replica['x_classification'] = replica['x_classification'].astype(str)
replica['x_classification'] = replica['x_classification'].map(lambda x: "X" + x)
replica
```

Out[21]:

| | rank | x_classification | start_datetime | max_datetime | end_datetime | region |
|---|---|---|---|---|---|---|
| **0** | 1 | X28.0 | 2003-11-04 20:00:00 | 2003-11-04 19:54:00 | 2003-11-04 00:00:00 | 10486 |
| **1** | 2 | X20.0 | 2001-04-02 22:05:00 | 2001-04-02 22:06:00 | 2001-04-03 02:30:00 | 9393 |
| **2** | 3 | X17.0 | 2003-10-28 11:10:00 | 2003-10-28 11:30:00 | 2003-10-29 00:00:00 | 10486 |
| **3** | 4 | X14.0 | 2001-04-15 14:05:00 | 2001-04-15 14:06:00 | 2001-04-16 13:00:00 | 9415 |
| **4** | 5 | X10.0 | 2003-10-29 20:55:00 | 2003-10-29 20:54:00 | 2003-10-29 00:00:00 | 10486 |
| **5** | 6 | X9.4 | 1997-11-06 12:20:00 | 1997-11-06 12:10:00 | 1997-11-07 08:30:00 | 8100 |
| **6** | 7 | X9.0 | 2006-12-05 10:50:00 | NaT | 2006-12-05 20:00:00 | 10930 |
| **7** | 8 | X8.3 | 2003-11-02 17:30:00 | 2003-11-02 17:30:00 | 2003-11-03 01:00:00 | 10486 |
| **8** | 9 | X7.1 | 2005-01-20 07:15:00 | 2005-01-20 06:54:00 | 2005-01-20 16:30:00 | 10720 |
| **9** | 10 | X6.9 | 2011-08-09 08:20:00 | 2011-08-09 08:12:00 | 2011-08-09 08:35:00 | 11263 |
| **10** | 11 | X6.5 | 2006-12-06 19:00:00 | NaT | 2006-12-08 00:00:00 | 10930 |
| **11** | 12 | X6.2 | 2005-09-09 19:45:00 | 2005-09-09 19:48:00 | 2005-09-09 22:00:00 | 10808 |
| **12** | 13 | X5.7 | 2000-07-14 10:30:00 | 2000-07-14 10:54:00 | 2000-07-15 14:30:00 | 9077 |
| **13** | 14 | X5.6 | 2001-04-06 19:35:00 | 2001-04-06 19:30:00 | 2001-04-07 01:50:00 | 9415 |
| **14** | 15 | X5.4 | 2012-03-07 01:00:00 | 2012-03-07 00:24:00 | 2012-03-08 19:00:00 | 11429 |
| **15** | 16 | X5.3 | 2001-08-25 16:50:00 | 2001-08-25 16:50:00 | 2001-08-25 23:00:00 | 9591 |
| **16** | 17 | X4.9 | 2014-02-25 00:56:00 | 2014-02-25 01:25:00 | 2014-02-25 11:28:00 | 11990 |
| **17** | 18 | X4.8 | 2002-07-23 00:50:00 | 2002-07-23 00:42:00 | 2002-07-23 04:00:00 | 10039 |
| **18** | 19 | X4.0 | 2000-11-26 17:00:00 | 2000-11-26 17:06:00 | 2000-11-26 17:15:00 | 9236 |
| **19** | 20 | X3.9 | 2003-11-03 10:00:00 | 2003-11-03 10:06:00 | 2003-11-03 12:30:00 | 10488 |
| **20** | 21 | X3.8 | 2005-01-17 10:00:00 | 2005-01-17 09:54:00 | 2005-01-17 10:35:00 | 10720 |
| **21** | 22 | X3.6 | 2003-05-28 01:00:00 | 2003-05-28 00:50:00 | 2003-05-29 00:30:00 | 10365 |
| **22** | 23 | X3.4 | 2001-12-28 20:35:00 | 2001-12-28 20:30:00 | 2001-12-29 03:00:00 | 9756 |
| **23** | 24 | X3.4 | 2006-12-13 02:45:00 | 2006-12-13 02:54:00 | 2006-12-13 10:40:00 | 10930 |
| **24** | 25 | X3.3 | 2002-07-20 21:30:00 | 2002-07-20 22:06:00 | 2002-07-20 22:20:00 | 10039 |
| **25** | 26 | X3.2 | 2013-05-14 01:16:00 | 2013-05-14 01:25:00 | 2013-05-14 02:35:00 | 11748 |
| **26** | 27 | X3.1 | 2002-08-24 01:45:00 | 2002-08-24 01:27:00 | 2002-08-24 03:25:00 | 10069 |
| **27** | 28 | X2.8 | 2013-05-13 16:15:00 | 2013-05-13 16:07:00 | 2013-05-13 19:10:00 | 11748 |
| **28** | 29 | X2.7 | 2003-11-03 01:15:00 | 2003-11-03 01:59:00 | 2003-11-03 01:25:00 | 10488 |
| **29** | 30 | X2.7 | 1998-05-06 08:25:00 | 1998-05-06 08:29:00 | 1998-05-06 08:35:00 | 8210 |
| **30** | 31 | X2.6 | 2005-01-15 23:00:00 | 2005-01-15 23:06:00 | 2005-01-15 00:00:00 | 10720 |
| **31** | 32 | X2.6 | 1997-11-27 13:30:00 | 1997-11-27 13:56:00 | 1997-11-27 14:00:00 | 8113 |
| **32** | 33 | X2.6 | 2001-09-24 10:45:00 | 2001-09-24 10:30:00 | 2001-09-25 20:00:00 | 9632 |
| **33** | 34 | X2.5 | 2004-11-10 02:25:00 | 2004-11-10 02:26:00 | 2004-11-10 03:40:00 | 10696 |
| **34** | 35 | X2.3 | 2001-04-10 05:24:00 | 2001-04-10 05:30:00 | 2001-04-10 00:00:00 | 9415 |
| **35** | 36 | X2.3 | 2000-06-06 15:20:00 | 2000-06-06 15:54:00 | 2000-06-08 09:00:00 | 9026 |
| **36** | 37 | X2.3 | 2000-11-24 15:25:00 | 2000-11-24 15:30:00 | 2000-11-24 22:00:00 | 9236 |
| **37** | 38 | X2.2 | 2011-02-15 02:10:00 | 2011-02-15 02:24:00 | 2011-02-15 07:00:00 | 11158 |
| **38** | 39 | X2.1 | 1997-11-04 06:00:00 | 1997-11-04 06:10:00 | 1997-11-05 04:30:00 | 8100 |
| **39** | 40 | X2.1 | 2005-09-10 21:45:00 | 2005-09-10 21:52:00 | 2005-09-10 01:00:00 | 10808 |
| **40** | 41 | X2.1 | 2011-09-06 22:30:00 | 2011-09-06 23:05:00 | 2011-09-07 15:40:00 | 11283 |
| **41** | 42 | X2.1 | 2013-10-25 15:08:00 | 2013-10-25 15:12:00 | 2013-10-25 22:32:00 | 11882 |
| **42** | 43 | X2.0 | 2001-04-12 10:20:00 | 2001-04-12 10:31:00 | 2001-04-12 10:40:00 | 9415 |
| **43** | 44 | X2.0 | 2005-01-17 09:25:00 | 2005-01-17 09:30:00 | 2005-01-17 16:00:00 | 10720 |
| **44** | 45 | X2.0 | 2000-11-24 05:10:00 | 2000-11-24 05:30:00 | 2000-11-24 15:00:00 | 9236 |
| **45** | 46 | X2.0 | 2004-11-07 16:25:00 | 2004-11-07 16:54:00 | 2004-11-08 20:00:00 | 10696 |
| **46** | 47 | X1.9 | 2000-11-25 19:00:00 | 2000-11-25 19:31:00 | 2000-11-25 19:35:00 | 9236 |
| **47** | 48 | X1.8 | 2002-07-18 07:55:00 | 2002-07-18 08:06:00 | 2002-07-18 08:45:00 | 10030 |
| **48** | 49 | X1.8 | 2000-11-24 22:24:00 | 2000-11-24 22:06:00 | 2000-11-24 22:36:00 | 9236 |
| **49** | 50 | X1.8 | 1999-10-14 09:10:00 | 1999-10-14 09:26:00 | 1999-10-14 10:00:00 | 8731 |

## Question 2: Integration

Here we want to match signals from the SpaceWeatherLive data with the NASA data. I used the processed NASA data from question 1 which will get rid of unnecessary things but it also keeps the data relatively the same. I added a new column called integrated_with_rank which gives us a value of 0 if it didnt find a correlating solar flare or a positive integer corresponding to the rank of the solar flare in the SpaceWeatherLive dataset. The way I determined the "best match" was by using the date and the region's last 3 digits as these are very unique values and are shared between the datasets.

In [22]:

```
#
nasa_integrated = replica
nasa_integrated['integrated with rank'] = '0'
for i, row_s in space.iterrows():
    for j, row_n in nasa_integrated.iterrows():
        if( (row_s['start_datetime'].date() == row_n['start_datetime'].date()) and (str(row_s['region'])[-3:] == str(row_n['region'])
[-3:]) ):
            nasa_integrated['integrated with rank'][j] = i+1

nasa_integrated
```

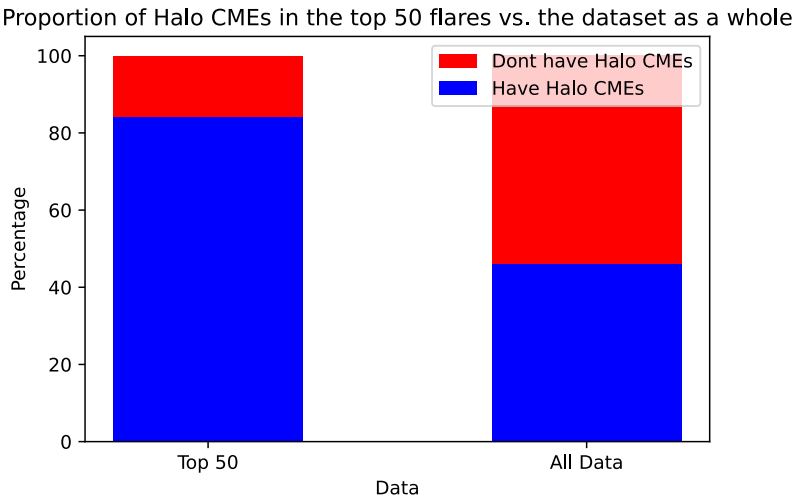
Out[22]:

| | rank | x_classification | start_datetime | max_datetime | end_datetime | region | integrated with rank |
|---|---|---|---|---|---|---|---|
| **0** | 1 | X28.0 | 2003-11-04 20:00:00 | 2003-11-04 19:54:00 | 2003-11-04 00:00:00 | 10486 | 1 |
| **1** | 2 | X20.0 | 2001-04-02 22:05:00 | 2001-04-02 22:06:00 | 2001-04-03 02:30:00 | 9393 | 2 |
| **2** | 3 | X17.0 | 2003-10-28 11:10:00 | 2003-10-28 11:30:00 | 2003-10-29 00:00:00 | 10486 | 3 |
| **3** | 4 | X14.0 | 2001-04-15 14:05:00 | 2001-04-15 14:06:00 | 2001-04-16 13:00:00 | 9415 | 5 |
| **4** | 5 | X10.0 | 2003-10-29 20:55:00 | 2003-10-29 20:54:00 | 2003-10-29 00:00:00 | 10486 | 6 |
| **5** | 6 | X9.4 | 1997-11-06 12:20:00 | 1997-11-06 12:10:00 | 1997-11-07 08:30:00 | 8100 | 7 |
| **6** | 7 | X9.0 | 2006-12-05 10:50:00 | NaT | 2006-12-05 20:00:00 | 10930 | 9 |
| **7** | 8 | X8.3 | 2003-11-02 17:30:00 | 2003-11-02 17:30:00 | 2003-11-03 01:00:00 | 10486 | 10 |
| **8** | 9 | X7.1 | 2005-01-20 07:15:00 | 2005-01-20 06:54:00 | 2005-01-20 16:30:00 | 10720 | 12 |
| **9** | 10 | X6.9 | 2011-08-09 08:20:00 | 2011-08-09 08:12:00 | 2011-08-09 08:35:00 | 11263 | 13 |
| **10** | 11 | X6.5 | 2006-12-06 19:00:00 | NaT | 2006-12-08 00:00:00 | 10930 | 14 |
| **11** | 12 | X6.2 | 2005-09-09 19:45:00 | 2005-09-09 19:48:00 | 2005-09-09 22:00:00 | 10808 | 31 |
| **12** | 13 | X5.7 | 2000-07-14 10:30:00 | 2000-07-14 10:54:00 | 2000-07-15 14:30:00 | 9077 | 17 |
| **13** | 14 | X5.6 | 2001-04-06 19:35:00 | 2001-04-06 19:30:00 | 2001-04-07 01:50:00 | 9415 | 18 |
| **14** | 15 | X5.4 | 2012-03-07 01:00:00 | 2012-03-07 00:24:00 | 2012-03-08 19:00:00 | 11429 | 19 |
| **15** | 16 | X5.3 | 2001-08-25 16:50:00 | 2001-08-25 16:50:00 | 2001-08-25 23:00:00 | 9591 | 22 |
| **16** | 17 | X4.9 | 2014-02-25 00:56:00 | 2014-02-25 01:25:00 | 2014-02-25 11:28:00 | 11990 | 23 |
| **17** | 18 | X4.8 | 2002-07-23 00:50:00 | 2002-07-23 00:42:00 | 2002-07-23 04:00:00 | 10039 | 0 |
| **18** | 19 | X4.0 | 2000-11-26 17:00:00 | 2000-11-26 17:06:00 | 2000-11-26 17:15:00 | 9236 | 26 |
| **19** | 20 | X3.9 | 2003-11-03 10:00:00 | 2003-11-03 10:06:00 | 2003-11-03 12:30:00 | 10488 | 47 |
| **20** | 21 | X3.8 | 2005-01-17 10:00:00 | 2005-01-17 09:54:00 | 2005-01-17 10:35:00 | 10720 | 29 |
| **21** | 22 | X3.6 | 2003-05-28 01:00:00 | 2003-05-28 00:50:00 | 2003-05-29 00:30:00 | 10365 | 33 |
| **22** | 23 | X3.4 | 2001-12-28 20:35:00 | 2001-12-28 20:30:00 | 2001-12-29 03:00:00 | 9756 | 0 |
| **23** | 24 | X3.4 | 2006-12-13 02:45:00 | 2006-12-13 02:54:00 | 2006-12-13 10:40:00 | 10930 | 34 |
| **24** | 25 | X3.3 | 2002-07-20 21:30:00 | 2002-07-20 22:06:00 | 2002-07-20 22:20:00 | 10039 | 0 |
| **25** | 26 | X3.2 | 2013-05-14 01:16:00 | 2013-05-14 01:25:00 | 2013-05-14 02:35:00 | 11748 | 39 |
| **26** | 27 | X3.1 | 2002-08-24 01:45:00 | 2002-08-24 01:27:00 | 2002-08-24 03:25:00 | 10069 | 0 |
| **27** | 28 | X2.8 | 2013-05-13 16:15:00 | 2013-05-13 16:07:00 | 2013-05-13 19:10:00 | 11748 | 43 |
| **28** | 29 | X2.7 | 2003-11-03 01:15:00 | 2003-11-03 01:59:00 | 2003-11-03 01:25:00 | 10488 | 47 |
| **29** | 30 | X2.7 | 1998-05-06 08:25:00 | 1998-05-06 08:29:00 | 1998-05-06 08:35:00 | 8210 | 48 |
| **30** | 31 | X2.6 | 2005-01-15 23:00:00 | 2005-01-15 23:06:00 | 2005-01-15 00:00:00 | 10720 | 49 |
| **31** | 32 | X2.6 | 1997-11-27 13:30:00 | 1997-11-27 13:56:00 | 1997-11-27 14:00:00 | 8113 | 0 |
| **32** | 33 | X2.6 | 2001-09-24 10:45:00 | 2001-09-24 10:30:00 | 2001-09-25 20:00:00 | 9632 | 50 |
| **33** | 34 | X2.5 | 2004-11-10 02:25:00 | 2004-11-10 02:26:00 | 2004-11-10 03:40:00 | 10696 | 0 |
| **34** | 35 | X2.3 | 2001-04-10 05:24:00 | 2001-04-10 05:30:00 | 2001-04-10 00:00:00 | 9415 | 0 |
| **35** | 36 | X2.3 | 2000-06-06 15:20:00 | 2000-06-06 15:54:00 | 2000-06-08 09:00:00 | 9026 | 0 |
| **36** | 37 | X2.3 | 2000-11-24 15:25:00 | 2000-11-24 15:30:00 | 2000-11-24 22:00:00 | 9236 | 0 |
| **37** | 38 | X2.2 | 2011-02-15 02:10:00 | 2011-02-15 02:24:00 | 2011-02-15 07:00:00 | 11158 | 0 |
| **38** | 39 | X2.1 | 1997-11-04 06:00:00 | 1997-11-04 06:10:00 | 1997-11-05 04:30:00 | 8100 | 0 |
| **39** | 40 | X2.1 | 2005-09-10 21:45:00 | 2005-09-10 21:52:00 | 2005-09-10 01:00:00 | 10808 | 0 |
| **40** | 41 | X2.1 | 2011-09-06 22:30:00 | 2011-09-06 23:05:00 | 2011-09-07 15:40:00 | 11283 | 0 |
| **41** | 42 | X2.1 | 2013-10-25 15:08:00 | 2013-10-25 15:12:00 | 2013-10-25 22:32:00 | 11882 | 0 |
| **42** | 43 | X2.0 | 2001-04-12 10:20:00 | 2001-04-12 10:31:00 | 2001-04-12 10:40:00 | 9415 | 0 |
| **43** | 44 | X2.0 | 2005-01-17 09:25:00 | 2005-01-17 09:30:00 | 2005-01-17 16:00:00 | 10720 | 29 |
| **44** | 45 | X2.0 | 2000-11-24 05:10:00 | 2000-11-24 05:30:00 | 2000-11-24 15:00:00 | 9236 | 0 |
| **45** | 46 | X2.0 | 2004-11-07 16:25:00 | 2004-11-07 16:54:00 | 2004-11-08 20:00:00 | 10696 | 0 |
| **46** | 47 | X1.9 | 2000-11-25 19:00:00 | 2000-11-25 19:31:00 | 2000-11-25 19:35:00 | 9236 | 0 |
| **47** | 48 | X1.8 | 2002-07-18 07:55:00 | 2002-07-18 08:06:00 | 2002-07-18 08:45:00 | 10030 | 0 |
| **48** | 49 | X1.8 | 2000-11-24 22:24:00 | 2000-11-24 22:06:00 | 2000-11-24 22:36:00 | 9236 | 0 |
| **49** | 50 | X1.8 | 1999-10-14 09:10:00 | 1999-10-14 09:26:00 | 1999-10-14 10:00:00 | 8731 | 0 |

## Question 3: Analysis

Here I decided to plot the proportions of Halo CMEs in the top 50 versus the rest of the data. From the results observed, there seemes to be a higher percentage of solar flares that have halo GMEs in the top 50 versus the percentage of solar flares in the rest of the data. This shows that solar flares with halo GMEs have a higher probability of being part of the top 50 solar flares. To calculate this, I used a function that returns all the rows that satisfy the condition of having the value "true". Witht that, I got the number of rows (which equals the number of solar flares) from the shape of the dataframe.

In [23]:
```
import matplotlib.pyplot as plt

top_50_count = nasa_top_50.loc[nasa_top_50['is_halo'] == True].shape[0]
nasa_count = nasa.loc[nasa['is_halo'] == True].shape[0]

nasa_count = nasa_count - top_50_count #remove the duplicates. nasa_count is also counting the ones in the top 50
top_50_count = top_50_count / nasa_top_50.shape[0] * 100
nasa_count = nasa_count / nasa.shape[0] * 100


countries = ['Top 50', 'All Data']
bottom = np.array([top_50_count, nasa_count])
top = np.array([100-top_50_count, 100-nasa_count])
idx = [x for x, _ in enumerate(countries)]

plt.bar(idx, top, width=0.5, label='Dont have Halo CMEs', color='red', bottom=bottom)
plt.bar(idx, bottom, width=0.5, label='Have Halo CMEs', color='blue')

plt.xticks(idx, countries)
plt.ylabel("Percentage")
plt.xlabel("Data")
plt.legend(loc="upper right")
plt.title("Proportion of Halo CMEs in the top 50 flares vs. the dataset as a whole")

plt.show()
```




In [ ]: